

TITLE OF THE INVENTION

SPEECH INFORMATION PROCESSING METHOD AND APPARATUS, AND
STORAGE MEDIUM

5

FIELD OF THE INVENTION

The present invention relates to a technique for synthesizing speech by using a speech segment dictionary.

10 BACKGROUND OF THE INVENTION

A speech synthesizing technique for synthesizing speech by using a computer uses a speech segment dictionary. This speech segment dictionary stores speech segments in units (synthetic units) of speech segments, CV/VC, or VCV.

15 To synthesize speech, appropriate speech segments are selected from this speech segment dictionary and modified and connected to generate desired synthetic speech. A flow chart in Fig. 15 explains this process.

In step S131, speech contents expressed by kana-kanji
20 mixed text and the like are input. In step S132, the input speech contents are analyzed to obtain a speech segment symbol string {p0, p1,...} and parameters for determining prosody. The flow then advances to step S133 to determine the prosody such as the speech segment time length,
25 fundamental frequency, and power. In speech segment dictionary look-up step S134, speech segments {w0, w1,...}

appropriate for the speech segment symbol string {p0, p1,...} obtained by the input analysis in step S132 and the prosody obtained by the prosody determination in step S133 are retrieved from the speech segment dictionary. The flow
5 advances to step S135, and the speech segments {w0, w1,...} obtained by the speech segment dictionary retrieval in step S134 are modified and concatenated to match the prosody determined in step S133. In step S136, the result of the speech segment modification and concatenation in step S135
10 is output as a synthetic speech.

Waveform editing is one effective method of speech synthesis. This method, e.g., superposes waveforms and changes pitches in synchronism with vocal cord vibrations. The method is advantageous in that synthetic speech close
15 to a natural utterance can be generated with a small amount of arithmetic operations. When a method like this is used, a speech segment dictionary is composed of indexes for retrieval, waveform data (also called speech segment data) corresponding to individual speech segments, and auxiliary
20 information of the data. In this case, all speech segment data registered in the speech segment dictionary are often encoded using the μ -law or ADPCM (Adaptive Differential Pulse Code Modulation).

The above prior art has the following problems.
25 First, when all speech segment data registered in the speech segment dictionary are encoded by using an encoding

09630355-080100

To achieve the above objects, a speech information processing method of the present invention is a speech information processing method of generating a speech segment dictionary for holding a plurality of speech segments, characterized by comprising the selection step of selecting an encoding method of encoding a speech segment from a plurality of encoding methods, the encoding step of encoding the speech segment by using the selected encoding method, and the storage step of storing the encoded speech segment in a speech segment dictionary.

A storage medium of the present invention is characterized by storing a control program for allowing a computer to realize the above speech information processing method.

15 A speech information processing apparatus of the present invention is a speech information processing apparatus for generating a speech segment dictionary for holding a plurality of speech segments, characterized by comprising selecting means for selecting an encoding method of encoding a speech segment from a plurality of encoding methods, encoding means for encoding the speech segment by using the selected encoding method, and storage means for storing the encoded speech segment in a speech segment dictionary.

25 A speech information processing method of the present invention is a speech information processing method of

synthesizing speech by using a speech segment dictionary for holding a plurality of speech segments, characterized by comprising the selection step of selecting, from a plurality of decoding methods, a decoding method of decoding a speech
5 segment read out from the speech segment dictionary, the decoding step of decoding the speech segment by using the selected decoding method, and the speech synthesizing step of synthesizing speech on the basis of the decoded speech segment.

10 A storage medium of the present invention is characterized by storing a control program for allowing a computer to realize the above speech information processing method.

A speech information processing apparatus of the
15 present invention is a speech information processing apparatus for synthesizing speech by using a speech segment dictionary for holding a plurality of speech segments, characterized by comprising selecting means for selecting, from a plurality of decoding methods, a decoding method of
20 decoding a speech segment read out from the speech segment dictionary, decoding means for decoding the speech segment by using the selected decoding method, and speech synthesizing means for synthesizing speech on the basis of the decoded speech segment.

25 A speech information processing method of the present invention is a speech information processing method of

generating a speech segment dictionary for holding a plurality of speech segments, characterized by comprising the setting step of setting an encoding method of encoding a speech segment in accordance with the type of the speech
5 segment, the encoding step of encoding the speech segment by using the set encoding method, and the storage step of storing the encoded speech segment in a speech segment dictionary.

A storage medium of the present invention is
10 characterized by comprising a control program for allowing a computer to realize the above speech information processing method.

A speech information processing apparatus of the present-invention is a speech information processing
15 apparatus for generating a speech segment dictionary for holding a plurality of speech segments, characterized by comprising setting means for setting an encoding method of encoding a speech segment in accordance with the type of the speech segment, encoding means for encoding the speech
20 segment by using the set encoding method, and storage means for storing the encoded speech segment in a speech segment dictionary.

A speech information processing method of the present invention is a speech information processing method of
25 synthesizing speech by using a speech segment dictionary for holding a plurality of speech segments, characterized by

comprising the setting step of setting a decoding method of
decoding a speech segment read out from the speech segment
dictionary in accordance with the type of the speech segment,
the decoding step of decoding the speech segment by using
5 the set decoding method, and the speech synthesizing step
of synthesizing speech on the basis of the decoded speech
segment.

A storage medium of the present invention is
characterized by comprising a control program for allowing
10 a computer to realize the above speech information processing
method.

A speech information processing apparatus of the
present invention is a speech information processing
apparatus for synthesizing speech by using a speech segment
15 dictionary for holding a plurality of speech segments,
characterized by comprising setting means for setting a
decoding method of decoding a speech segment read out from
the speech segment dictionary in accordance with the type
of the speech segment, decoding means for decoding the speech
20 segment by using the set decoding method, and speech
synthesizing means for synthesizing speech on the basis of
the decoded speech segment.

Other features and advantages of the present invention
will be apparent from the following description taken in
25 conjunction with the accompanying drawings, in which like

reference characters designate the same or similar parts throughout the figures thereof.

BRIEF DESCRIPTION OF THE DRAWINGS

5 The accompanying drawings, which are incorporated in and constitute a part of the specification, illustrate embodiments of the invention and, together with the description, serve to explain the principles of the invention.

10 Fig. 1 is block diagram showing the hardware configuration of a speech synthesizing apparatus according to each embodiment of the present invention;

15 Fig. 2 is a flow chart for explaining a speech segment dictionary formation algorithm in the first embodiment of the present invention;

 Fig. 3 is a flow chart for explaining a speech synthesis algorithm in the first embodiment of the present invention;

20 Fig. 4 is a flow chart for explaining a speech segment dictionary formation algorithm in the second embodiment of the present invention;

 Fig. 5 is a flow chart for explaining a speech synthesis algorithm in the second embodiment of the present invention;

25 Fig. 6 is a flow chart for explaining a speech segment dictionary formation algorithm in the third embodiment of the present invention;

Q

09630356-080100

Fig. 7 is a flow chart for explaining the speech segment dictionary formation algorithm in the third embodiment of the present invention;

Fig. 8 is a flow chart for explaining a speech synthesis
5 algorithm in the third embodiment of the present invention;

Fig. 9 is a flow chart for explaining a speech segment dictionary formation algorithm in the fourth embodiment of the present invention;

Fig. 10 is a flow chart for explaining a speech
10 synthesis algorithm in the fourth embodiment of the present invention;

Fig. 11 is a flow chart for explaining a speech segment dictionary formation algorithm in the fifth embodiment of the present invention;

Fig. 12 is a flow chart for explaining a speech
15 synthesis algorithm in the fifth embodiment of the present invention;

Fig. 13 is a flow chart for explaining a speech segment dictionary formation algorithm in the sixth embodiment of
20 the present invention;

Fig. 14 is a flow chart for explaining a speech synthesis algorithm in the sixth embodiment of the present invention; and

Fig. 15 is a flow chart showing a general speech
25 synthesizing process.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

Preferred embodiments of the present invention will be described in detail below with reference to the accompanying drawings. In these embodiments, (1) a method
5 of forming a speech segment dictionary (a speech segment dictionary formation algorithm) and (2) a method of synthesizing speech by using this speech segment dictionary (a speech synthesis algorithm) will be described in detail.

Fig. 1 is a block diagram showing an outline of the
10 functional configuration of a speech information processing apparatus according to the embodiments of the present invention. A speech segment dictionary formation algorithm and a speech synthesis algorithm in each embodiment are realized by using this speech information processing
15 apparatus.

Referring to Fig. 1, a central processing unit (CPU)
100 executes numerical operations and various control processes and controls operations of individual units (to be described later) connected via a bus 105. A storage device
20 101 includes, e.g., a RAM and ROM and stores various control programs executed by the CPU 100, data, and the like. The storage device 101 also temporarily stores various data necessary for the control by the CPU 100. An external storage device 102 is a hard disk device or the like and includes
25 speech segment database 111 and a speech segment dictionary 112. This speech segment database 111 holds speech segments

before registration in the speech segment dictionary 112 (i.e., non-compressed speech segments). An output device 103 includes a monitor for displaying the operation statuses of diverse programs, a loudspeaker for outputting
5 synthesized speech, and the like. An input device 104 includes, e.g., a keyboard and a mouse. By using this input device 104, a user can control a program for forming the speech segment dictionary 112, control a program for synthesizing speech by using the speech segment dictionary 112, and input
10 text (containing a plurality of character strings) as an object of speech synthesis.

On the basis of the above configuration, a speech segment dictionary formation algorithm and a speech synthesis algorithm in each embodiment will be described
15 below.

[First Embodiment]

A speech segment dictionary formation algorithm and a speech synthesis algorithm according to the first embodiment of the present invention will be described below
20 by using the speech processing apparatus shown in Fig. 1.

In the first embodiment, one of a plurality of encoding methods (more specifically, a 7-bit μ -law scheme and an 8-bit μ -law scheme) different in the number of quantization steps is selected for each speech segment to be registered in a
25 speech segment dictionary 112. Note that a speech segment to be registered in the speech segment dictionary 112 is

composed of a phoneme, semi-phoneme, diphone (e.g., CV or VC), VCV (or CVC), or combinations thereof.

(Formation of speech segment dictionary)

Fig. 2 is a flow chart for explaining the speech segment
5 dictionary formation algorithm in the first embodiment of
the present invention. A program for achieving this
algorithm is stored in a storage device 101. A CPU 100 reads
out this program from the storage device 101 on the basis
of an instruction from a user and executes the following
10 procedure.

In step S201, the CPU 100 initializes an index i, which indicates each of N speech segment data (each speech segment data is non-compressed) stored in speech segment database 111 of an external storage device 102, to "0". Note that this
15 index i is stored in the storage device 101.

In step S202, the CPU 100 reads out i th speech segment data W_i indicated by this index \underline{i} . Assume that the readout data W_i is

$$W_i = \{x_0, x_1, \dots, x_{T-1}\}$$

20 where T is the time length (in units of samples) of W_i .

In step S203, the CPU 100 encodes the speech segment data W_i read out in step S202 by using the 7-bit μ -law scheme. Assume that the result of the encoding is

$$C_i = \{c_0, c_1, \dots, c_{T-1}\}$$

25 In step S204, the CPU 100 calculates encoding
distortion ρ produced by the 7-bit μ -law encoding in step

12

S203. In this embodiment, a mean square error ρ is used as a measure of this encoding distortion. This mean square error ρ can be represented by

$$\rho = (1/T) \cdot \sum (x_t - \mu(t))^{-1} (c_t))^2 \quad \dots (1)$$

5 where $\mu(7)^{-1}()$ is a 7-bit μ -law decoding function. In this equation, " Σ " is the summation from $t = 0$ to $t = T - 1$.

In step S205, the CPU 100 checks whether the encoding distortion ρ calculated in step S204 is larger than a predetermined threshold value ρ_0 . If $\rho > \rho_0$, the CPU 100 determines that the waveform of the speech segment data W_i is distorted by encoding using the 7-bit μ -law scheme. Therefore, in step S206 the CPU 100 switches the encoding method to the 8-bit μ -law scheme having a different number of quantization bits. In other cases, the flow advances to step S207. In step S206, the CPU 100 encodes the speech segment data W_i read out in step S202 by using the 8-bit μ -law scheme. Assume that the result of the encoding is

$$C_i = \{c_0, c_1, \dots, c_{T-1}\}$$

In step S207, the CPU 100 writes encoding information
20 of the phoneme data W_i and the like in the phoneme dictionary
112. In addition to the encoding information, the CPU 100
writes information necessary to decode the phoneme data W_i .
This encoding information specifies the encoding method by
which the speech segment data W_i is encoded:

25 The encoding information is "0" if the encoding method
is the 7-bit μ -law scheme

The encoding information is "1" if the encoding method is the 8-bit μ -law scheme

In step S208, the CPU 100 writes the speech segment data W_i encoded by one encoding scheme in the speech segment dictionary 112. In step S209, the CPU 100 checks whether the
5 above processing is performed for all of the N speech segment data. If $i = N - 1$, the CPU 100 completes this algorithm. If not, in step S210 the CPU 100 adds 1 to the index i , the flow returns to step S202, and the CPU 100 reads out speech
10 segment data designated by the updated index i . The CPU 100 repeatedly executes this processing for all of the N speech segment data.

In the speech segment dictionary formation algorithm of the first embodiment as described above, an encoding
15 scheme can be selected from the 7-bit μ -law scheme and the 8-bit μ -law scheme for each speech segment to be registered in the speech segment dictionary 112. With this arrangement, a storage capacity necessary for the speech segment
dictionary can be very efficiently reduced without
20 deteriorating the quality of speech segments to be registered in the speech segment dictionary. Also, a larger number of types of speech segments than in conventional speech segment dictionaries can be registered in a speech segment dictionary having a storage capacity equivalent to those of the
25 conventional dictionaries.

In the first embodiment, the aforementioned speech segment dictionary formation algorithm is realized on the basis of the program stored in the storage device 101. However, a part or the whole of this speech segment dictionary formation algorithm can also be constituted by hardware.

(Speech synthesis)

Fig. 3 is a flow chart for explaining the speech synthesis algorithm in the first embodiment of the present invention. A program for achieving this algorithm is stored in the storage device 101. The CPU 100 reads out this program on the basis of an instruction from a user and executes the following procedure.

In step S301, the user inputs a character string in Japanese, English, or some other language by using the keyboard and the mouse of an input device 104. In the case of Japanese, the user inputs a character string expressed by kana-kanji mixed text. In step S302, the CPU 100 analyzes the input character string and obtains the speech segment sequence of this character string and parameters for determining the prosody of this character string. In step S303, on the basis of the prosodic parameters obtained in step S302, the CPU 100 determines prosody such as a duration length (the prosody for controlling the length of a voice), fundamental frequency (the prosody for controlling the pitch of a voice), and power (the prosody for controlling the strength of a voice).

In step S304, the CPU 100 obtains an optimum speech segment sequence on the basis of the speech segment sequence obtained in step S302 and the prosody determined in step S303. The CPU 100 selects one speech segment contained in this speech segment sequence and retrieves speech segment data corresponding to the selected speech segment and encoding information corresponding to this speech segment data. If the speech segment dictionary 112 is stored in a storage medium such as a hard disk, the CPU 100 sequentially seeks to storage areas of encoding information and speech segment data. If the speech segment dictionary 112 is stored in a storage medium such as a RAM, the CPU 100 sequentially moves a pointer (address register) to storage areas of encoding information and speech segment data.

In step S305, the CPU 100 reads out the encoding information retrieved in step S304 from the speech segment dictionary 112. This encoding information indicates the encoding method of the speech segment data retrieved in step S304:

If the encoding information is "0", the encoding method is the 7-bit μ -law scheme

If the encoding information is "1", the encoding method is the 8-bit μ -law scheme

In step S306, the CPU 100 examines the encoding information read out in step S305. If the encoding information is "0", the CPU 100 selects a decoding method

In the speech synthesis algorithm of the first embodiment as described above, a desired speech segment can be decoded by a decoding method corresponding to the 7-bit μ -law scheme or the 8-bit μ -law scheme. With this arrangement, natural, high-quality synthetic speech can be generated.

In the first embodiment, the aforementioned speech synthesis algorithm is realized on the basis of the program stored in the storage device 101. However, a part or the whole of this speech synthesis algorithm can also be constituted by hardware.

[First Modification of the First Embodiment]

In the first embodiment, speech segment data whose encoding distortion is larger than a predetermined threshold value is encoded by the 8-bit μ -law scheme. However, it is also possible to obtain the encoding distortion after encoding is performed by the 8-bit μ -law scheme, and register speech segment data whose encoding distortion is larger than a predetermined threshold value in a speech segment dictionary without encoding the data. With this arrangement, degradation of the quality of an unstable speech segment (e.g., a speech segment classified into a voiced fricative sound or a plosive) can be prevented. Also, natural, high-quality synthetic speech can be generated by using a speech segment dictionary thus formed.

[Second Modification of the First Embodiment]

007080"950356
09630356 080100

In the first embodiment, an encoding method is selected from the 7-bit μ -law scheme and the 8-bit μ -law scheme in accordance with the encoding distortion. However, it is also possible, in accordance with the type (e.g., a voiced
5 fricative sound, plosive, nasal sound, some other voiced sound, or unvoiced sound) of speech segment, to choose to encode the speech segment by the 7-bit μ -law scheme or the 8-bit μ -law scheme or to register the speech segment in the speech segment dictionary 112 without encoding it. For
10 example, a speech segment of the type of a voiced fricative sound and plosive may be registered in the speech segment dictionary 112 without encoding it, and a speech segment of the type of nasal sound and unvoiced sound may be registered in the speech segment dictionary 112 by encoding with the
15 7-bit μ -law scheme, and a speech segment of the type of other voiced sound may be registered in the speech segment dictionary 112 by encoding with the 8-bit μ -law scheme.
[Second Embodiment]

20 A speech segment dictionary formation algorithm and a speech synthesis algorithm according to the second embodiment of the present invention will be described below by using the speech processing apparatus shown in Fig. 1.

In the second embodiment, one of a plurality of encoding methods using different quantization code books is
25 selected for each speech segment to be registered in a speech segment dictionary 112. Note that a speech segment to be

registered in the speech segment dictionary 112 is composed of a phoneme, semi-phoneme, diphone (e.g., CV or VC), VCV (or CVC), or combinations thereof.

(Formation of speech segment dictionary)

5 Fig. 4 is a flow chart for explaining the speech segment dictionary formation algorithm in the second embodiment of the present invention. A program for achieving this algorithm is stored in a storage device 101. A CPU 100 reads out this program from the storage device 101 on the basis
10 of an instruction from a user and executes the following procedure.

 In step S401, the CPU 100 initializes an index i , which indicates each of N speech segment data (each speech segment data is non-compressed) stored in speech segment database
15 111 of an external storage device 102, to "0". Note that this index i is stored in the storage device 101.

 In step S402, the CPU 100 reads out i th speech segment data W_i indicated by this index i . Assume that the readout data W_i is

20 $W_i = \{x_0, x_1, \dots, x_{T-1}\}$

where T is the time length (in units of samples) of W_i .

 In step S403, the CPU 100 forms a scalar quantization code book Q_i of the speech segment data W_i read out in step S402. More specifically, the CPU 100 decodes the encoded
25 speech segment data W_i by using the scalar quantization code book Q_i and so designs that a mean square error ρ of decoded

data sequence $Y_i = \{y_0, y_1, \dots, y_{T-1}\}$ is a minimum (i.e., the encoding distortion is a minimum). In this case, an algorithm such as an LBG method is usable. With this arrangement, the distortion of the waveform of a speech segment produced by encoding can be minimized. Note that the mean square error ρ can be represented by

$$\rho = (1/T) \cdot \sum (x_t - y_t)^2 \quad \dots (2)$$

where " Σ " is the summation from $t = 0$ to $t = T - 1$.

In step S404, the CPU 100 writes the scalar quantization code book Q_i formed in step S403 and the like in the speech segment dictionary 112. In addition to the quantization code book Q_i , the CPU 100 writes information necessary to decode the speech segment data W_i . In step S405, the CPU 100 encodes (scalar-quantizes) the speech segment data W_i by using the quantization code book Q_i formed in step S403.

Assuming the code book Q_i is

$$Q_i = \{q_0, q_1, \dots, q_{N-1}\} \quad (N \text{ is the quantization step}),$$

a code c_t corresponding to x_t ($\in W_i$) can be represented by

$$c_t = \text{argn min } (x_t - q_n)^2 \quad (0 \leq n < N) \quad \dots (3)$$

In step S406, the CPU 100 writes speech segment data C_i ($= \{c_0, c_1, \dots, c_{T-1}\}$ encoded in step S405 into the speech segment dictionary 112. In step S407, the CPU 100 checks whether the above processing is performed for all of the N speech segment data. If $i = N - 1$, the CPU 100 completes this algorithm. If not, in step S408 the CPU 100 adds 1 to the

invention. A program for achieving this algorithm is stored in the storage device 101. The CPU 100 reads out this program on the basis of an instruction from a user and executes the following procedure.

5 In step S501, the user inputs a character string in Japanese, English, or some other language by using the keyboard and the mouse of an input device 104. In the case of Japanese, the user inputs a character string expressed by kana-kanji mixed text. In step S502, the CPU 100 analyzes
10 the input character string and obtains the speech segment sequence of this character string and parameters for determining the prosody of this character string. In step S503, on the basis of the prosodic parameters obtained in step S502, the CPU 100 determines prosody such as a duration
15 length (the prosody for controlling the length of a voice), fundamental frequency (the prosody for controlling the pitch of a voice), and power (the prosody for controlling the strength of a voice).

In step S504, the CPU 100 obtains an optimum speech
20 segment sequence on the basis of the speech segment sequence obtained in step S502 and the prosody determined in step S503. The CPU 100 selects one speech segment contained in this speech segment sequence and retrieves a scalar quantization code book and speech segment data corresponding to the
25 selected speech segment. If the speech segment dictionary 112 is stored in a storage medium such as a hard disk, the

001080" 950350

CPU 100 sequentially seeks to storage areas of scalar quantization code books and speech segment data. If the speech segment dictionary 112 is stored in a storage medium such as a RAM, the CPU 100 sequentially moves a pointer
5 (address register) to storage areas of scalar quantization code books and speech segment data.

In step S505, the CPU 100 reads out the scalar quantization code book retrieved in step S504 from the speech segment dictionary 112. In step S506, the CPU 100 reads out
10 the speech segment data retrieved in step S504 from the speech segment dictionary 112. In step S507, the CPU 100 decodes the speech segment data read out in step S506 by using the scalar quantization code book read out in step S505.

In step S508, the CPU 100 checks whether speech segment
15 data corresponding to all speech segments contained in the speech segment sequence obtained in step S504 are decoded. If all speech segment data are decoded, the flow advances to step S509. If speech segment data not decoded yet is present, the flow returns to step S504 to decode the next
20 speech segment data.

In step S509, on the basis of the prosody determined in step S503, the CPU 100 modifies and connects the decoded speech segments (i.e., edits the waveform). In step S510, the CPU 100 outputs the synthetic speech obtained in step
25 S509 from the loudspeaker of an output device 103.

09630356 "080100

In the speech synthesis algorithm of the second embodiment as described above, a desired speech segment can be decoded using an optimum quantization code book for the speech segment. Accordingly, natural, high-quality
5 synthetic speech can be generated.

In the second embodiment, the aforementioned speech synthesis algorithm is realized on the basis of the program stored in the storage device 101. However, a part or the whole of this speech synthesis algorithm can also be
10 constituted by hardware.

[First Modification of the Second Embodiment]

In the second embodiment, as in the first embodiment described previously, the number of bits (i.e., the number of quantization steps of scalar quantization) per sample can
15 be changed for each speech segment data. This can be accomplished by changing the procedures of the second embodiment as follows. That is, in the speech segment dictionary formation algorithm, the number of quantization steps is determined prior to the process (the write of the
20 scalar quantization code book) in step S404 of Fig. 4. The determined number of quantization steps and the code book are recorded in the speech segment dictionary 112. In the speech synthesis algorithm, the number of quantization steps is read out from the speech segment dictionary 112 before
25 the process (the read-out of the scalar quantization code book) in step S505. As in the first embodiment, the number

of quantization steps can be determined on the basis of the encoding distortion.

[Second Modification of the Second Embodiment]

In the speech synthesis algorithm of the second
5 embodiment, in step S505 a scalar quantization code book
formed for each speech segment data is selected. However,
the present invention is not limited to this embodiment. For
example, from a plurality of types of scalar quantization
code books previously held by the speech segment dictionary
10 112, a code book having the highest performance (i.e., by
which the quantization distortion is a minimum) can also be
chosen.

[Third Modification of the Second Embodiment]

In the second embodiment, a quantization code book is
15 so designed that the encoding distortion is a minimum, and
speech segment data is scalar-quantized by using the designed
quantization code book. However, speech segment data whose
encoding distortion is larger than a predetermined threshold
value can also be registered in a speech segment dictionary
20 without being encoded. With this arrangement, degradation
of the quality of an unstable speech segment (e.g., a speech
segment classified into a voiced fricative sound or a
plosive) can be prevented. Also, natural, high-quality
synthetic speech can be generated by using a speech segment
25 dictionary thus formed.

[Third Embodiment]

A speech segment dictionary formation algorithm and a speech synthesis algorithm according to the second embodiment of the present invention will be described below by using the speech processing apparatus shown in Fig. 1.

5 In the above second embodiment, one of a plurality of
encoding methods using different quantization code books is
selected for each speech segment to be registered in a speech
segment dictionary 112. In this third embodiment, however,
one of a plurality of encoding methods using different
10 quantization code books is selected for each of a plurality
of speech segment clusters. Note that a speech segment to
be registered in the speech segment dictionary 112 is
composed of a phoneme, semi-phoneme, diphone (e.g., CV or
VC), VCV (or CVC), or combinations thereof.

15 (Formation of speech segment dictionary)

Fig. 6 is a flow chart for explaining the speech segment dictionary formation algorithm in the third embodiment of the present invention. A program for achieving this algorithm is stored in a storage device 101. A CPU 100 reads out this program from the storage device 101 on the basis of an instruction from a user and executes the following procedure.

In step S601, the CPU 100 reads out all of N speech
segment data (each speech segment data is non-compressed)
25 stored in speech segment database 111 of an external storage
device 102. In step S602, the CPU 100 clusters all these

speech segments into a plurality of (M) speech segment clusters. More specifically, the CPU 100 forms M speech segment clusters in accordance with the similarity of the waveform of each speech segment.

5 In step S603, the CPU 100 initializes index i which indicates each of the M speech segment clusters to "0". In step S604, the CPU 100 forms a scalar quantization code book Q_i for i th speech segment cluster L_i . In step S605, the CPU 100 writes the code book Q_i formed in step S604 into the speech
10 segment dictionary 112.

 In step S606, the CPU 100 checks whether the above processing is performed for all of the M speech segment clusters. If $i = M - 1$ (the processing is completely performed for all of the M speech segment clusters), the flow
15 advances to step S608. If not, in step S607 the CPU 100 adds 1 to the index i , the flow returns to step S604, and the CPU 100 forms a scalar quantization code book for the next speech segment cluster.

 After scalar quantization code books are formed for
20 all of the M speech segment clusters, this algorithm advances to step S608. In step S608, the CPU 100 initializes index i , which indicates each of the N speech segments stored in the speech segment database 111 of the external storage device 102, to "0". In step S609, the CPU 100 selects a scalar
25 quantization code book Q_i for i th speech segment data W_i . This scalar quantization code book Q_i selected is a

quantization code book corresponding to a speech segment cluster to which the speech segment data W_i belongs.

In step S610, the CPU 100 writes information (code book information) designating the scalar quantization code book selected in step S609 and the like into the speech segment dictionary 112. In addition to the code book information, the CPU 100 writes information necessary to decode the speech segment data W_i . In step S611, the CPU 100 encodes the speech segment data W_i by using the code book Q_i formed in step S604.

10 In step S612, the CPU 100 writes speech segment data C_i ($= \{c_0, c_1, \dots, c_{T-1}\}$ encoded in step S611 into the speech segment dictionary 112.

In step S613, the CPU 100 checks whether the above processing is performed for all of the N speech segment data.

15 If $i = N - 1$, the CPU 100 completes this algorithm. If not, in step S614 the CPU 100 adds 1 to the index i , the flow returns to step S609, and the CPU 100 forms a scalar quantization code book for the next speech segment data.

In the speech segment dictionary formation algorithm of the third embodiment as described above, one of a plurality of encoding methods using different quantization code books can be selected for each of a plurality of speech segment clusters. This can reduce the number of quantization code books to be registered in the speech segment dictionary 112.

20

25 With this arrangement, a storage capacity necessary for the speech segment dictionary can be very efficiently reduced

without deteriorating the quality of speech segments to be registered in the speech segment dictionary. Also, a larger number of types of speech segments than in conventional speech segment dictionaries can be registered in a speech segment dictionary having a storage capacity equivalent to those of the conventional dictionaries.

In the third embodiment, the aforementioned speech segment dictionary formation algorithm is realized on the basis of the program stored in the storage device 101.

However, a part or the whole of this speech segment dictionary formation algorithm can also be constituted by hardware. (Speech synthesis)

Fig. 8 is a flow chart for explaining the speech synthesis algorithm in the third embodiment of the present invention. A program for achieving this algorithm is stored in the storage device 101. The CPU 100 reads out this program on the basis of an instruction from a user and executes the following procedure. For the sake of simplicity, in this embodiment it is assumed that code books corresponding to all speech segment clusters are previously stored in the storage device 101.

Steps S801 to 803 have the same functions and processes as in steps S501 to S503 of Fig. 5, so a detailed description thereof will be omitted.

In step S804, the CPU 100 obtains an optimum speech segment sequence on the basis of a speech segment sequence

obtained in step S802 and prosody determined in step S803. The CPU 100 selects one speech segment contained in this speech segment sequence and retrieves code book information and speech segment data corresponding to the selected speech segment. If the speech segment dictionary 112 is stored in a storage medium such as a hard disk, the CPU 100 sequentially seeks to storage areas of code book information and speech segment data. If the speech segment dictionary 112 is stored in a storage medium such as a RAM, the CPU 100 sequentially moves a pointer (address register) to storage areas of code book information and speech segment data.

In step S805, the CPU 100 reads out the code book information retrieved in step S804 and determines a speech segment cluster of this speech segment data and a scalar quantization code book corresponding to the speech segment cluster. In step S806, the CPU 100 looks up the speech segment dictionary 112 to obtain the scalar quantization code book determined in step S805. In step S807, the CPU 100 reads out the speech segment data retrieved in step S804 from the speech segment dictionary 112. In step S808, the CPU 100 decodes the speech segment data read out in step S807 by using the scalar quantization code book obtained in step S806.

In step S809, the CPU 100 checks whether speech segment data corresponding to all speech segments contained in the speech segment sequence obtained in step S804 are decoded. If all speech segment data are decoded, the flow advances

to step S810. If speech segment data not decoded yet is present, the flow returns to step S804 to decode the next speech segment data.

In step S810, on the basis of the prosody determined
5 in step S803, the CPU 100 modifies and connects the decoded
speech segments (i.e., edits the waveform). In step S811,
the CPU 100 outputs the synthetic speech obtained in step
S810 from the loudspeaker of an output device 103.

In the speech synthesis algorithm of the third embodiment as described above, a desired speech segment can be decoded using an optimum quantization code book for a speech segment cluster to which this speech segment belongs. Accordingly, natural, high-quality synthetic speech can be generated.

15 In the third embodiment, the aforementioned speech synthesis algorithm is realized on the basis of the program stored in the storage device 101. However, a part or the whole of this speech synthesis algorithm can also be constituted by hardware.

20 [First Modification of the Third Embodiment]

In the speech segment dictionary formation algorithm of the third embodiment, the procedure of forming a speech segment cluster in accordance with the similarity of the waveform of a speech segment has been explained. However, 25 it is also possible to form a speech segment cluster in accordance with the type (e.g., a voiced fricative sound,

plosive, nasal sound, some other voiced sound, or unvoiced sound) of speech segment, and form a quantization code book for each speech segment cluster.

[Second Modification of the Third Embodiment]

5 In the speech synthesis algorithm of the third embodiment, in step S805 a scalar quantization code book formed for each speech segment cluster is selected. However, the present invention is not limited to this embodiment. For example, from a plurality of types of scalar quantization
10 code books held by the speech segment dictionary 112, a code book having the highest performance (i.e., by which the quantization distortion is a minimum) can also be chosen.

[Third Modification of the Third Embodiment]

 In the third embodiment, scalar quantization can also
15 be performed by taking the gain (power) into consideration. That is, in step 609 a gain g of speech segment data is obtained prior to selecting a scalar quantization code book. In step S610, the obtained gain g and code book information are written in the speech segment dictionary 112. In step S611,
20 quantization is performed by taking account of the gain g . This means that equation (3) presented earlier is replaced by

$$ct = \arg \min (x_t - g \cdot q_n)^2 \quad (0 \leq n < N)$$

 Meanwhile, in step S808 (reference to a code book) of
25 the speech synthesis algorithm, the value g obtained by the

code book reference is multiplied by the gain g to yield a decoded value.

[Fourth Modification of the Third Embodiment]

In the third embodiment, an optimum quantization code
5 book is designed for each speech segment cluster, and speech
segment data belonging to each speech segment cluster is
scalar-quantized by using the designed quantization code
book. However, speech segment data found to increase the
encoding distortion can also be registered in a speech
10 segment dictionary without being encoded. With this
arrangement, degradation of the quality of an unstable speech
segment (e.g., a speech segment classified into a voiced
fricative sound or a plosive) can be prevented. Also,
natural, high-quality synthetic speech can be generated by
15 using a speech segment dictionary thus formed.

[Fourth Embodiment]

A speech segment dictionary formation algorithm and
a speech synthesis algorithm according to the fourth
embodiment of the present invention will be described below
20 by using the speech processing apparatus shown in Fig. 1.

In the fourth embodiment, a linear prediction
coefficient and a prediction difference are calculated for
each speech segment data, and the data is encoded by an optimum
quantization code book for the calculated prediction
25 difference. Note that a speech segment to be registered in
the speech segment dictionary 112 is composed of a phoneme,

semi-phoneme, diphone (e.g., CV or VC), VCV (or CVC), or combinations thereof.

(Formation of speech segment dictionary)

Fig. 9 is a flow chart for explaining the speech segment
5 dictionary formation algorithm in the fourth embodiment of
the present invention. A program for achieving this
algorithm is stored in a storage device 101. A CPU 100 reads
out this program from the storage device 101 on the basis
of an instruction from a user and executes the following
10 procedure.

In step S901, the CPU 100 initializes an index i , which indicates each of N speech segment data (each speech segment data is non-compressed) stored in speech segment database 111 of an external storage device 102, to "0". In step S902, the CPU 100 reads out speech segment data (a speech segment before encoding) W_i of the i th speech segment indicated by this index i . Assume that the readout data W_i is

$$W_i = \{x_0, x_1, \dots, x_{T-1}\}$$

where T is the time length (in units of samples) of W_i .

20 In step S903, the CPU 100 calculates a linear
prediction coefficient and a prediction difference of the
speech segment data W_i read out in step S902. Assuming the
linear prediction order is order L , this linear prediction
model is represented by using a linear prediction coefficient
25 a_l and a prediction difference dt as

$$x_t = \sum a_l x_{t-1} + d_t \quad \dots (4)$$

where Σ is the summation of $l = 1$ to L .

Hence, the linear prediction coefficient a_1 which minimizes the square-sum of the prediction difference d_t

$$\Sigma dt^2 \quad \dots (5)$$

5 is determined. In this expression, Σ is the summation of
t = 1 to T - 1.

In step S904, the CPU 100 writes the linear prediction coefficient a_l calculated in step S903 into the speech segment dictionary 112. In step S905, the CPU 100 forms a quantization code book Q_i of the prediction difference d_t calculated in step S903. More specifically, the CPU 100 decodes the encoded prediction difference d_t by using the quantization code book Q_i and so designs that a mean square error ρ of decoded data sequence $E_i = \{e_1, e_{1+1}, \dots, e_{T-1}\}$ is a minimum (i.e., the encoding distortion is a minimum). In this case, an algorithm such as an LBG method is usable. With this arrangement, the distortion of the waveform of a speech segment produced by encoding can be minimized. Note that the mean square error ρ can be represented by

$$\rho = (1/T) \cdot \sum (dt - et)^2 \quad \dots (6)$$

where " Σ " is the summation of $t = 0$ to $T - 1$.

In step S906, the CPU 100 writes the quantization code book Q_i formed in step S905 and the like in the speech segment dictionary 112. In addition to the code book Q_i , the CPU 100 writes information necessary to decode the speech segment data W_i . In step S907, the CPU 100 encodes the speech segment

data W_i by linear predictive coding by using the linear prediction coefficient a_l calculated in step S903 and the code book Q_i formed in step S905. Assuming the code book Q_i is

5 $Q_i = \{q_0, q_1, \dots, q_{N-1}\}$ (N is the quantization step),
a code c_t corresponding to x_t ($\in W_i$) can be represented by
$$c_t = \text{argn min } (x_t - \sum a_l y_{t-1} - q_n)^2 \quad (0 \leq n < N)$$

... (7)

where y_t is the value obtained by encoding and then decoding
10 x_t by this method.

In step S908, the CPU 100 writes speech segment data C_i ($= \{c_0, c_1, \dots, c_{T-1}\}$ encoded in step S907 into the speech segment dictionary 112. In step S909, the CPU 100 checks whether the above processing is performed for all of the N
15 speech segment data. If $i = N - 1$, the CPU 100 completes this algorithm. If not, in step S910 the CPU 100 adds 1 to the index i , the flow returns to step S902, and the CPU 100 reads out speech segment data designated by the updated index i . The CPU 100 repeatedly executes this processing for all of
20 the N speech segment data.

In the speech segment dictionary formation algorithm of the fourth embodiment as described above, it is possible to calculate a linear prediction coefficient and a prediction difference for each speech segment to be registered in the
25 speech segment dictionary 112, and encode the speech segment by an optimum quantization code book for the calculated

prediction difference. With this arrangement, a storage capacity necessary for the speech segment dictionary can be very efficiently reduced without deteriorating the quality of speech segments to be registered in the speech segment dictionary. Also, a larger number of types of speech segments than in conventional speech segment dictionaries can be registered in a speech segment dictionary having a storage capacity equivalent to those of the conventional dictionaries.

In the fourth embodiment, the aforementioned speech segment dictionary formation algorithm is realized on the basis of the program stored in the storage device 101. However, a part or the whole of this speech segment dictionary formation algorithm can also be constituted by hardware.

(Speech synthesis)

Fig. 10 is a flow chart for explaining the speech synthesis algorithm in the fourth embodiment of the present invention. A program for achieving this algorithm is stored in the storage device 101. The CPU 100 reads out this program on the basis of an instruction from a user and executes the following procedure.

In step S1001, the user inputs a character string in Japanese, English, or some other language by using the keyboard and the mouse of an input device 104. In the case of Japanese, the user inputs a character string expressed by kana-kanji mixed text. In step S1002, the CPU 100 analyzes

the input character string and obtains the speech segment sequence of this character string and parameters for determining the prosody of this character string. In step S1003, on the basis of the prosodic parameters obtained in step S1002, the CPU 100 determines prosody such as a duration length (the prosody for controlling the length of a voice), the fundamental frequency (the prosody for controlling the pitch of a voice), and the power (the prosody for controlling the strength of a voice).

10 In step S1004, the CPU 100 obtains an optimum speech
segment sequence on the basis of the speech segment sequence
obtained in step S1002 and the prosody determined in step
S1003. The CPU 100 selects one speech segment contained in
this speech segment sequence and retrieves a linear
15 prediction coefficient, quantization code book, and
prediction difference corresponding to the selected speech
segment. If the speech segment dictionary 112 is stored in
a storage medium such as a hard disk, the CPU 100 sequentially
seeks to storage areas of linear prediction coefficients,
20 quantization code books, and prediction differences. If the
speech segment dictionary 112 is stored in a storage medium
such as a RAM, the CPU 100 sequentially moves a pointer
(address register) to storage areas of linear prediction
coefficients, quantization code books, and prediction
25 differences.

15

20

25

40

06-01-00

09630356 "030100

In step S1005, the CPU 100 reads out the prediction coefficient retrieved in step S1004 from the speech segment dictionary 112. In step S1006, the CPU 100 reads out the quantization code book retrieved in step S1004 from the
5 speech segment dictionary 112. In step S1007, the CPU 100 reads out the prediction difference retrieved in step S1004 from the speech segment dictionary 112. In step S1008, the CPU 100 decodes the prediction difference by using the prediction coefficient, the quantization code book, and the
10 decoded data of the immediately preceding sample, thereby obtaining speech segment data.

In step S1009, the CPU 100 checks whether speech segment data corresponding to all speech segments contained in the speech segment sequence obtained in step S1004 are
15 decoded. If all speech segment data are decoded, the flow advances to step S1010. If speech segment data not decoded yet is present, the flow returns to step S1004 to decode the next speech segment data.

In step S1010, on the basis of the prosody determined
20 in step S1003, the CPU 100 modifies and connects the decoded speech segments (i.e., edits the waveform). In step S1011, the CPU 100 outputs the synthetic speech obtained in step S1010 from the loudspeaker of an output device 103.

In the speech synthesis algorithm of the fourth
25 embodiment as described above, a desired speech segment can be decoded using an optimum quantization code book for the

speech segment. Accordingly, natural, high-quality synthetic speech can be generated.

In the fourth embodiment, the aforementioned speech synthesis algorithm is realized on the basis of the program stored in the storage device 101. However, a part or the whole of this speech synthesis algorithm can also be constituted by hardware.

[First Modification of the Fourth Embodiment]

In the fourth embodiment, as in the first embodiment described earlier, the number of bits (i.e., the number of quantization steps) per sample can be changed for each speech segment data. This can be accomplished by changing the procedures of the fourth embodiment as follows. That is, in the speech segment dictionary formation algorithm, the number of quantization steps is determined prior to the process (the write of the quantization code book) in step S905. The determined number of quantization steps and the code book are recorded in the speech segment dictionary 112. In the speech synthesis algorithm, the number of quantization steps is read out from the speech segment dictionary 112 before the process (the read-out of the quantization code book) in step S1006. As in the first embodiment, the number of quantization steps can be determined on the basis of the encoding distortion.

[Second Modification of the Fourth Embodiment]

In the fourth embodiment, one quantization code book is designed for one speech segment data. However, one quantization code book can also be designed for a plurality of speech segment data. For example, as in the third
5 embodiment, it is possible to cluster N speech segment data into M speech segment clusters and design a quantization code book for each speech segment cluster.

[Fifth Modification of the Fourth Embodiment]

In the fourth embodiment, data of L samples from the
10 beginning of speech segment data can be directly written in the speech segment dictionary 112 without being encoded. This makes it possible to avoid a phenomenon in which linear prediction cannot be well performed for L samples from the beginning of speech segment data.

15 [Sixth Modification of the Fourth Embodiment]

In the fourth embodiment, in step S907 the code ct that is optimum for xt is obtained. However, this optimum code ct can also be obtained by taking account of m samples after xt. This can be realized by temporarily determining the code
20 ct and recursively searching for the code ct (searching the tree structure).

[Seventh Modification of the Fourth Embodiment]

In the fourth embodiment, a quantization code book is so designed that the encoding distortion is a minimum, and
25 speech segment data is linearly encoded by using the designed quantization code book. However, speech segment data whose

09630355 "080100
encoding distortion is larger than a predetermined threshold value can be registered in a speech segment dictionary without being encoded. With this arrangement, degradation of the quality of an unstable speech segment (e.g., a speech
5 segment classified into a voiced fricative sound or a plosive) can be prevented. Also, natural, high-quality synthetic speech can be generated by using a speech segment dictionary thus formed.

[Fifth Embodiment]

10 A speech segment dictionary formation algorithm and a speech synthesis algorithm according to the fifth embodiment of the present invention will be described below by using the speech processing apparatus shown in Fig. 1.

In the fifth embodiment, the various encoding schemes
15 used in the previous embodiments are combined, and an optimum encoding method is selected for each speech segment data to be registered in a speech segment dictionary 112. In this fifth embodiment, an unstable speech segment (e.g., a speech segment classified into a voiced fricative sound or a
20 plosive) is processed without being compressed. Note that a speech segment to be registered in the speech segment dictionary 112 is composed of a phoneme, semi-phoneme, diphone (e.g., CV or VC), VCV (or CVC), or combinations thereof.

25 (Formation of speech segment dictionary)

CPU 100 encodes the speech segment data W_i by using another encoding scheme. If $\rho > \rho_0$ does not hold, the flow advances to step S1106.

In step S1106, the CPU 100 writes encoding information of the speech segment data W_i in the speech segment dictionary 112. This encoding information contains information specifying the encoding method by which the speech segment data W_i is encoded and information necessary to decode the speech segment data W_i (e.g., a prediction coefficient and a quantization code book). In step S1107, the CPU 100 writes the speech segment data W_i encoded in step S1103 into the speech segment dictionary 112, and the flow advances to step S1120.

On the other hand, in step S1108 the CPU 100 encodes the speech segment data W_i read out in step S1102 by using the encoding scheme (i.e., the 7-bit μ -law scheme or the 8-bit μ -law scheme) explained in the first embodiment.

In step S1109, the CPU 100 calculates encoding distortion ρ by this encoding scheme. In step S1110, the CPU 100 checks whether the encoding distortion ρ calculated in step S1109 is larger than a predetermined threshold value ρ_1 . If $\rho > \rho_1$, the flow advances to step S1113, and the CPU 100 encodes the speech segment data W_i by using another encoding scheme. If $\rho > \rho_1$ does not hold, the flow advances to step S1111.

In step S1111, the CPU 100 writes encoding information of the speech segment data W_i in the speech segment dictionary 112. This encoding information contains information specifying the encoding method by which the speech segment data W_i is encoded and information necessary to decode the speech segment data W_i . In step S1112, the CPU 100 writes the speech segment data W_i encoded in step S1108 into the speech segment dictionary 112, and the flow advances to step S1120.

10 On the other hand, in step S1113 the CPU 100 encodes
the speech segment data W_i read out in step S1102 by using
the encoding scheme (i.e., scalar quantization) explained
in the second or third embodiment.

In step S1114, the CPU 100 calculates encoding distortion ρ by this encoding scheme. In step S1115, the CPU 100 checks whether the encoding distortion ρ calculated in step S1114 is larger than a predetermined threshold value ρ_2 . For example, the waveform of a strongly unstable speech segment (e.g., a speech segment classified into a voiced fricative sound or a plosive) largely varies, so $\rho > \rho_2$ does not hold. If $\rho > \rho_2$, the flow advances to step S1118. If $\rho > \rho_2$ does not hold, the flow advances to step S1116.

In step S1116, the CPU 100 writes encoding information of the speech segment data W_i in the speech segment dictionary 112. This encoding information contains information specifying the encoding method by which the speech segment

48

data W_i is encoded and information necessary to decode the speech segment data W_i (e.g., a quantization code book). In step S1117, the CPU 100 writes the speech segment data W_i encoded in step S1113 into the speech segment dictionary 112, and the flow advances to step S1120.

On the other hand, in step S1118 the CPU 100 writes encoding information of the speech segment data W_i read out in step S1102 into the speech segment dictionary 112 without compressing the speech segment data W_i . This encoding information contains information indicating that the speech segment data W_i is not encoded. In step S1119, the CPU 100 writes this speech segment data W_i in the speech segment dictionary 112, and the flow advances to step S1120. With this arrangement, deterioration of the quality of an unstable speech segment can be prevented.

In step S1120, the CPU 100 checks whether the above processing is performed for all of the N speech segment data. If $i = N - 1$, the CPU 100 completes this algorithm. If not, in step S1121 the CPU 100 adds 1 to the index i , the flow returns to step S1102, and the CPU 100 reads out speech segment data designated by the updated index i . The CPU 100 repeatedly executes this processing for all of the N speech segment data.

In the speech segment dictionary formation algorithm of the fifth embodiment as described above, an encoding scheme can be selected from the μ -law scheme, scalar

quantization, and linear predictive coding for each speech segment to be registered in the speech segment dictionary 112. With this arrangement, a storage capacity necessary for the speech segment dictionary can be very efficiently reduced without deteriorating the quality of speech segments to be registered in the speech segment dictionary. Also, a larger number of types of speech segments than in conventional speech segment dictionaries can be registered in a speech segment dictionary having a storage capacity equivalent to those of the conventional dictionaries.

In the fifth embodiment, the aforementioned speech segment dictionary formation algorithm is realized on the basis of the program stored in the storage device 101. However, a part or the whole of this speech segment dictionary formation algorithm can also be constituted by hardware. (Speech synthesis)

Fig. 12 is a flow chart for explaining the speech synthesis algorithm in the fifth embodiment of the present invention. A program for achieving this algorithm is stored in the storage device 101. The CPU 100 reads out this program on the basis of an instruction from a user and executes the following procedure.

In step S1201, the user inputs a character string in Japanese, English, or some other language by using the keyboard and the mouse of an input device 104. In the case of Japanese, the user inputs a character string expressed

speech segment data retrieved in step S1204 from the speech segment dictionary 112.

In step S1207, on the basis of the encoding information read out in step S1205, the CPU 100 checks whether the speech
5 segment data read out in step S1206 is encoded. If the data is encoded, the flow advances to step S1208 to specify the encoding method. If the data is not encoded, the flow advances to step S1215.

In step S1208, on the basis of the encoding information
10 read out in step S1205, the CPU 100 examines the encoding method of the speech segment data read out in step S1206. If the encoding method is linear predictive coding, the flow advances to step S1212 to decode the data. In other cases, the flow advances to step S1209.

In step S1209, on the basis of the encoding information
15 read out in step S1205, the CPU 100 examines the encoding method of the speech segment data read out in step S1206. If the encoding method is the μ -law scheme, the flow advances to step S1213 to decode the data. In other cases, the flow
20 advances to step S1210.

In step S1210, on the basis of the encoding information read out in step S1205, the CPU 100 examines the encoding method of the speech segment data read out in step S1206. If the encoding method is scalar quantization, the flow
25 advances to step S1214 to decode the data. In other cases, the flow advances to step S1211.

In step S1211, the CPU 100 checks whether speech segment data corresponding to all speech segments contained in the speech segment sequence obtained in step S1204 are decoded. If all speech segment data are decoded, the flow advances to step S1215. If speech segment data not decoded yet is present, the flow returns to step S1204 to decode the next speech segment data.

In step S1215, on the basis of the prosody determined in step S1203, the CPU 100 modifies and connects the decoded speech segments (i.e., edits the waveform). In step S1216, the CPU 100 outputs the synthetic speech obtained in step S1215 from the loudspeaker of an output device 103.

In the speech synthesis algorithm of the fifth embodiment as described above, a desired speech segment can be decoded by a decoding method corresponding to one of the μ -law scheme, scalar quantization, and linear predictive coding. Therefore, natural, high-quality synthetic speech can be generated.

In the fifth embodiment, the aforementioned speech
20 synthesis algorithm is realized on the basis of the program
stored in the storage device 101. However, a part or the
whole of this speech synthesis algorithm can also be
constituted by hardware.

[Sixth Embodiment]

25 A speech segment dictionary formation algorithm and
a speech synthesis algorithm according to the sixth

embodiment of the present invention will be described below by using the speech processing apparatus shown in Fig. 1.

In the above fifth embodiment, an optimum encoding method is selected from a plurality of encoding methods using different encoding schemes for each speech segment data to be registered in a speech segment dictionary 112. In the sixth embodiment, however, an optimum encoding method is chosen from a plurality of encoding methods using different encoding schemes in accordance with the type of speech segment data. Note that a speech segment to be registered in the speech segment dictionary 112 is constructed of a phoneme, semi-phoneme, diphone (e.g., CV or VC), VCV (or CVC), or combinations thereof.

(Formation of speech segment dictionary)

Fig. 13 is a flow chart for explaining the speech segment dictionary formation algorithm in the sixth embodiment of the present invention. A program for achieving this algorithm is stored in a storage device 101. A CPU 100 reads out this program from the storage device 101 on the basis of an instruction from a user and executes the following procedure.

In step S1301, the CPU 100 initializes an index i , which indicates each of N speech segment data (each speech segment data is non-compressed) stored in speech segment database 111 of an external storage device 102, to "0". Note that this index i is stored in the storage device 101.

In step S1302, the CPU 100 reads out i th speech segment data W_i indicated by this index i . Assume that the readout data W_i is

$$W_i = \{x_0, x_1, \dots, x_{T-1}\}$$

5 where T is the time length (in units of samples) of W_i .

In step S1303, the CPU 100 discriminates the type of the speech segment data W_i read out in step S1302. More specifically, the CPU 100 checks whether the type of the speech segment data W_i is a voiced fricative sound, plosive,
10 unvoiced sound, nasal sound, or some other voiced sound.

If the type of the speech segment data W_i is a voiced fricative sound or plosive, the flow advances to step S1316. In step S1316, the CPU 100 does not compress this speech segment data W_i . With this arrangement, degradation of the
15 quality of the voiced fricative sound or plosive can be prevented. In step S1316, the CPU 100 writes encoding information of the speech segment data W_i in the speech segment dictionary 112. This encoding information contains
20 the type of the speech segment data W_i and information indicating that the speech segment data W_i is not encoded. In step S1317, the CPU 100 writes the speech segment data W_i in the speech segment dictionary 112 without encoding the speech segment data W_i , and the flow advances to step S1318.

If the type of the speech segment data is an unvoiced
25 sound, the flow advances to step S1306. In step S1306, the CPU 100 encodes the speech segment data W_i by using the

encoding scheme (i.e., scalar quantization) explained in the second or third embodiment. In step S1307, the CPU 100 writes encoding information of the speech segment data W_i in the speech segment dictionary 112. This encoding information contains the type of the speech segment data W_i , information specifying the encoding method by which the speech segment data W_i is encoded, and information necessary to decode the speech segment data W_i (e.g., a quantization code book). In step S1308, the CPU 100 writes the speech segment data W_i encoded in step S1306 into the speech segment dictionary 112, and the flow advances to step S1318.

If the type of the speech segment data is a nasal sound, the flow advances to step S1310. In step S1310, the CPU 100 encodes the speech segment data W_i by using the encoding scheme (i.e., linear predictive coding) explained in the fourth embodiment. In step S1311, the CPU 100 writes encoding information of the speech segment data W_i in the speech segment dictionary 112. This encoding information contains the type of the speech segment data W_i , information specifying the encoding method by which the speech segment data W_i is encoded, and information necessary to decode the speech segment data W_i (e.g., a prediction coefficient and a quantization code book). In step S1312, the CPU 100 writes the speech segment data W_i encoded in step S1310 into the speech segment dictionary 112, and the flow advances to step S1318.

09630355 030100

If the type of the speech segment data W_i is some other voiced sound, the flow advances to step S1313. In step S1313, the CPU 100 encodes the speech segment data W_i by using the encoding scheme (i.e., the 7-bit μ -law scheme or the 8-bit μ -law scheme) explained in the first embodiment. In step S1314, the CPU 100 writes encoding information of the speech segment data W_i in the speech segment dictionary 112. This encoding information contains the type of the speech segment data W_i , information specifying the encoding method by which the speech segment data W_i is encoded, and information necessary to decode the speech segment data W_i . In step S1315, the CPU 100 writes the speech segment data W_i encoded in step S1313 into the speech segment dictionary 112, and the flow advances to step S1318.

In step S1318, the CPU 100 checks whether the above processing is performed for all of the N speech segment data. If $i = N - 1$, the CPU 100 completes this algorithm. If not, in step S1319 the CPU 100 adds 1 to the index i , the flow returns to step S1302, and the CPU 100 reads out speech segment data designated by the updated index i . The CPU 100 repeatedly executes this processing for all of the N speech segment data.

In the speech segment dictionary formation algorithm of the sixth embodiment as described above, an encoding scheme can be selected from the μ -law scheme, scalar quantization, and linear predictive coding in accordance

Sub
B4

Sub
B5

Sub
B6

20

25

in the speech segment sequence obtained in step S1404 are decoded. If all speech segment data are decoded, the flow advances to step S1418. If speech segment data not decoded yet is present, the flow returns to step S1404 to decode the
5 next speech segment data.

In step S1418, on the basis of the prosody determined in step S1403, the CPU 100 modifies and connects the decoded speech segments (i.e., edits the waveform). In step S1419, the CPU 100 outputs the synthetic speech obtained in step 10 S1418 from the loudspeaker of an output device 103.

In the speech synthesis algorithm of the sixth embodiment as described above, a desired speech segment can be decoded by a decoding method corresponding to one of the μ -law scheme, scalar quantization, and linear predictive coding. With this arrangement, natural, high-quality synthetic speech can be generated.

In the sixth embodiment, the aforementioned speech synthesis algorithm is realized on the basis of the program stored in the storage device 101. However, a part or the whole of this speech synthesis algorithm can also be constituted by hardware.

[Other Embodiments]

In the second, fourth, and fifth embodiments described above, scalar quantization is used as the method of quantization. However, vector quantization can also be

61

applied by regarding a plurality of consecutive samples as one vector.

Also, it is possible to divide an unstable speech segment such as a plosive into two portions before and after
5 the plosion and encode these two portions by their respective optimum encoding methods. This can further improve the encoding efficiency of an unstable speech segment.

The fourth embodiment has been explained on the basis of a linear prediction model. However, some other vocal cord
10 filter model is also applicable. For example, an LMA (Log Magnitude Approximation) filter coefficient can be used in place of a linear prediction coefficient, and model parameters can be calculated by using the residual error of this LMA filter instead of a prediction difference. With
15 this arrangement, the fourth embodiment can be applied to the cepstrum domain.

Each of the above embodiments is applicable to a system comprising a plurality of devices (e.g., a host computer, interface device, reader, and printer) or to an apparatus
20 (e.g., a copying machine or facsimile apparatus) comprising a single device.

In each of the above embodiments, on the basis of instructions by program codes read out by the CPU 100, an operating system (OS) or the like running on the CPU 100 can
25 execute a part or the whole of actual processing.

Furthermore, in each of the above embodiments, program codes read out from the storage device 101 are written in a memory of a function extension unit connected to the CPU 100, and a CPU or the like of this function extension unit
5 executes a part or the whole of actual processing on the basis of instructions by the program codes.

In each of the embodiments as described above, an encoding method can be selected for each speech segment data. Therefore, a storage capacity necessary for the speech
10 segment dictionary can be very efficiently reduced without deteriorating the quality of speech segments to be registered in the speech segment dictionary. Also, natural, high-quality synthetic speech can be generated by using the speech segment dictionary thus formed.

15 The present invention is not limited to the above embodiments and various changes and modifications can be made within the spirit and scope of the present invention. Therefore, to apprise the public of the scope of the present invention, the following claims are made.